Available on CMS information server

CMS NOTE 1998/087

The Compact Muon Solenoid Experiment

CMS Note

Mailing address: CMS CERN, CH-1211 GENEVA 23, Switzerland

December 1, 1998

Getting Physics Data From the CMS ECAL Construction Database

J.-M. Le Goff, I. Willers

CERN, Geneva, Switzerland

R. McClatchey, Z. Kovacs

Centre for Complex Cooperative Systems, Univ. West of England, Frenchay, Bristol BS16 1QY UK

F. Martin, F. Zach

IPNL, IN2P3, Villeurbanne, Lyon

L. Dobrzynski

Ecole Polytechnic, IN2P3, Paris

Abstract

CMS ECAL physicists must be able to extract physics characteristics from the ECAL construction database for the calibration of sets of detector components. Other applications, such as geometry for simulation and physics event reconstruction, will also need to extract data from the construction database. In each case, application software needs to query the construction database and to extract data that satisfies a particular view. These *viewpoints* are defined for a specific purpose (e.g. simulation, slow control, calibration) and data must be extracted into the viewpoint for a set of defined detector components (e.g. readout channels) called ‘physics elements’.

The ECAL construction database follows an object-oriented design to maximise flexibility and reusability. An meta-modelling approach has been taken in its design, which promotes self-description and a degree of data independence. A query facility is being provided to allow navigation around so-called ‘meta-objects’ in the construction database, facilitating the extraction of physics data into a particular viewpoint. This paper outlines how viewpoints can be populated with data extracted from the construction database, for a set of detector elements relevant for analysis.

1. Calibration Data and ECAL Construction

CMS ECAL physicists will be able to collect all of the salient characteristics of each ECAL component, as it is characterised and assembled in a construction database, CRISTAL ([1], [2], [3]), from the end of 1998. Following assembly of CMS, it is essential for event recognition programs, amongst other applications, to have access to detector characteristics in order for calibration and event reconstruction to take place. Therefore the physics data collected during the assembly phase must be arranged, processed and presented to calibration physicists in a manner that facilitates population of any (pre-)calibration database.

ECAL will produce a pre-calibration database to hold the information required for calculation of the final calibration constants on a per-supermodule basis. However, construction and calibration are two quite different *views* of the detector data. Consequently, physicists must be given facilities to extract sets of data, relevant for determining calibration constants, from the ECAL construction database for storage in the pre-calibration database. The information required by the ECAL Barrel for calibration includes data for crystals, capsules, supermodules and electronics. Crystal-specific information will include light yields, attenuation lengths, longitudinal and transverse Transmission and any crystal non-uniformities. Capsule information needed for calibration includes photo-current and gain and dark current vs. high voltage. All of this information will be captured in the construction database as the detector is constructed step-wise from individual crystals to sub-units, modules and super-modules.

The step-wise construction procedure leads to a data organisation which is necessarily different from that required for calibration: the structure of the construction database follows the assembly ordering of the calorimeter, while the structure of the calibration database must follow the ECAL readout structure, where the unit of detector that is normally considered is the readout channel. Calibration constants need to be determined for each readout channel (e.g. a crystal plus its capsule, electronics (ADC etc.) and optical fibres) so that ADC counts can be translated into energy deposited in a single readout channel. In essence, the calibration system must be able to *extract* subsets of physics characteristics from the construction database for the calibration of particular physics elements (or sets of detector components) even if these elements are specified in a manner which is different to that in which structures are defined in the ECAL construction database.

In other words, the tree-representation of the ECAL construction database, established during assembly, must be traversed and physics characteristics extracted for calibration-sensitive components. All of the parameters needed for the calibration of a single supermodule must be extracted from the ECAL construction database and a matrix, {i,j}, of readout channels needs to be built in the pre-calibration database, as shown in figure 1. The construction database tree structure is navigated for a set of user selected ‘physics elements’, in this case readout channels, which define a calibration *viewpoint* onto the construction database. This concept of viewpoints and physics elements is pursued further in a later section of this paper.

To build a complete picture of the conditions under which calibration data is taken, information from test beam slow controls, data acquisition and monitoring is added to the physics data extracted from the construction database and, on completion of the calibration runs, these data are copied back into secure central storage (see figure 2). The central storage then accumulates this pre-calibration data for each supermodule and acts as the source of calibration data for all 36 final supermodules. The static calibration constants for the complete ECAL are finally extracted from the central storage into a final calibration database. Fast and efficient access from physicist programmes must be allowed both to the pre-calibration database for a single supermodule, at that time in the H4 test beam, and to the set of constants in the full calibration database (see figure 2).

To facilitate the implementation of the navigation and data extraction tools required, the construction database is object-oriented in nature and is being designed around so-called *meta-objects* [4] (see figure 3). This design will provide isolation of any modification to the database from any software accessing it. The extraction facilities are being designed to be sufficiently generic in nature to be used as the basis for ad-hoc database queries in ECAL physics analysis.

This paper is organised as follows: the next section of this paper justifies the meta-object approach to CRISTAL database design. Section 3 generalises this approach and identifies how a CMS-wide meta-model can be used to provide multiple viewpoint access for physicists to the data held in the construction database. Having discussed the concepts of 'physics elements' and viewpoints, section 4 revisits the ECAL calibration as a concrete example and investigates in detail the use of viewpoints by physicists for building a pre-calibration database for individual ECAL supermodules from the data captured in CRISTAL during supermodule assembly and testing.

2. The Need for Self-Description and Meta-Objects

The scale of LHC and its experiments increasingly requires the use of industrial-strength systems for data management to cope with system complexity. As the HEP construction process evolves, so more data, and the corresponding relationships between different aspects of the data, must be permanently recorded. HEP groups require flexible ways to find, access and share this construction data. The actual information required will depend on the viewpoint adopted by, and the role of, the user in the organisation. HEP user groups may well require a maintenance, a survey or an experiment systems management viewpoint in addition to a calibration viewpoint as detailed in the previous section. Also, over time, new distributed computing systems will need to interoperate with the CRISTAL repository in, as yet, unforseeable ways.

One important aspect in providing for viewpoints and interoperability concerns ways of making components and systems *self describing*. Systems should be designed to be able to retain knowledge about their dynamic structure (and how this structure has been accessed) and for this knowledge to be available to the rest of the distributed infrastructure through *the way that the system is plugged together*. This is absolutely critical and necessary for the next generation of HEP systems to be able to cope with the size and complexity explosion.

There is no doubt that as the CMS construction process gets underway production schemes and part specifications will continue to evolve. Clearly, these changes in definition must be folded into any data which is derived from the construction data system (one example is the pre-calibration system, as outlined in the previous section). To cope with this a production management system must, ideally, be able to support dynamic self-reconfiguration. One way of achieving this is for the system to make available a representation of itself for manipulation. A system which can make modifications to itself by virtue of its own computation is called a reflective system [5]. In order to inter-operate in an environment of future systems and in order to adapt to reconfigurations and versions of itself, large HEP systems should become self-describing or reflective. The representation needed for self-description is often termed meta-data. In CRISTAL a concept of *meta-objects* is introduced to provide for interoperability, flexibility required by HEP groups and to reduce system complexity.

A meta-object is defined for each class of significance in the CRISTAL data model: e.g part definitions for parts, activity definitions for activities, and agents definitions for agents. Figure 3 shows the part definition meta-object. In the model information is stored for types of parts or part definitions as well as for individual instantiations of parts. At the design stage of the project, information is stored against the definition object (design produces an ‘as-designed’ model) and only when assembly of a part has been completed is information stored on an individual part basis (the ‘as-built’ model). This meta-object approach reduces apparent system complexity by promoting object reuse and by translating complex hierarchies of object instances into graphs of object definitions. Meta-objects allow the capture of knowledge (about the object) alongside the object themselves, enriching the model and facilitating self-description and data independence. It is believed that the use of meta-objects provides the flexibility needed to cope with the evolution of definitions over the extended timescales of detector production and the flexibility required to cope with ad-hoc activity specification.

In constructing the CRISTAL data model, the Unified Modeling language (UML) [6] methodology of Booch, Rumbaugh & Jacobson has been followed; the result being a detailed UML model, presented elsewhere [7]. A simplified subset of the CRISTAL meta model is shown in figure 4. This model describes relationships, types, inheritance, containment and other associations between the meta objects in the system. The meta-objects in the model are definitions (for example part definitions or activity definitions) and the definitions are either elementary or composite in nature. CompositeMember objects capture the membership of objects in other objects. The data description world of, in this example, parts and the process description world of, in this case activities, displays an elegant symmetry with respect to compositeness. Figure 4 shows that there is an association between a given activity meta-object definition and a named part definition meta-object. The CRISTAL data model has been designed so that each assignment of a Part Definition to an Activity Definition is declared for a specific purpose. For the purpose of detector construction, the assignment is made to indicate the process to be instantiated for the assembly of a particular instance of a part, of a given part definition. Each assignment has associated with it some *conditions*: in detector construction, the data model captures the definition of the conditions required for each assignment of an activity definition to a part definition.

This technique can be generalised for other applications. For example, the association of a maintenance activity to a part will require quite different conditions to be captured than when the detector was constructed. Also, the association of a calibration activity to a part would require calibration-specific conditions to be captured. In other words, the identified association between the process and part description worlds carries rich semantics. This method allows for the integration of a Product Data Management-view of the detector and a Workflow Management-view of the detector through the definition of meta-objects [2] and their mutual assignment is very powerful. It allows many other links to be made between aspects of the overall CRISTAL data model: the same mechanism can be used to assign agents to activity definitions for the purposes of enactment or the assignment of agents to part definitions for the purposes of resource management.

3. Generalising Physicist Access to CRISTAL Data

The calculation of calibration coefficients for particular readout channels is just one viewpoint from which the ECAL construction database needs to be consulted for characteristics which have been gathered during assembly and testing. In this example the physics element definition of interest is a readout channel, which is the basic unit of the calibration system. Other applications will also need to extract sets of viewpoint-specific data e.g. for control and monitoring of the equipment, for alignment, geometry for simulation and, ultimately, for physics reconstruction programmes and, in each case, sets of physics elements can be defined for the purpose. The application software will need to traverse the construction database for each viewpoint in which physics characteristics are required, in order to extract data into a viewpoint-specific repository for the physics elements as specified by a physicist. This is true even if the structure of the physics elements does not necessarily follow the structures defined in the construction database. For example in the Upper Level Readout of ECAL Barrel, so-called Trigger Towers are defined as units of 5 readout strips (each of 5 crystals) – this 5 by 5 representation exists nowhere in the construction database. The physicist defines a viewpoint in terms of ‘physics elements’ (sets of detector components) which are derived from the tree of physical locations of detector components. A viewpoint is simply a set of physics elements of the same definition defined for the capture of data (i.e. ADC content) specific to the viewpoint. Figure 1 shows that a matrix of physics elements can be extracted from the ‘as-designed’ detector construction graph by providing software (meta-queries) which can traverse the construction tree and can query and extract the construction data for a selected set of detector components.

Meta-modelling can assist the extraction of data for physics elements provided such a *query* mechanism is developed which can navigate the meta-model, can interpret the structures in the meta-model and can present the data in a form meaningful to the end-user. The meta-model (or detector ‘as-designed’) is used to specify a required viewpoint in terms of a ‘physics element’ definition: a calibration viewpoint, for example, uses a readout channel definition which itself refers to crystal, capsule and electronics part definition meta-objects. The 'as-designed' graph is used to declare the type of physics element required for the viewpoint and the graph is then navigated to build the corresponding actual physics elements following their actual physical location in the hardware. The physics element definitions are captured in the meta-model and stored for reuse by other applications (see figure 5) and to allow correlation between viewpoints. One clear example is the use of readout channels as physics elements: readout channels are re-used by multiple applications (e.g beam events, monitoring events) and are therefore constituents of multiple viewpoints. Over time the meta-model will hold the definitions of multiple viewpoints (i.e multiple physics element definitions) and will consequently record all details of viewpoint usage.

A viewpoint (such as a calibration viewpoint) is therefore constructed by instantiating a set of ‘physics elements’ from their corresponding physics element definition (see figure 6). The query or data extraction facility comprises a set of software processes (or so-called Agents) which can be invoked either by a viewpoint-specific application (e.g. calibration) or by a viewpoint non-specific application. The agents either navigate a generalised meta-model to project out viewpoint-specific data (i.e. ‘looking in’ the meta-model) or they navigate the meta-model to correlate effects between separate viewpoints (i.e. ‘looking out’ from the data model).

In the ‘looking in’ (viewpoint-specific) case the agents perform the traversal of the detector description, following selected physics elements in the construction graph and extracing the relevant physics characteristics for the application. (This being the case of the pre-calibration example of section 1). In the ‘looking out’ case (viewpoint non-specific), the agents are used to determine the effect of a system-wide change on individual viewpoints or sets of viewpoints i.e an effect across viewpoints. As an example of inter-viewpoint navigation consider a request to determine the effects of a detector temperature variation: a change in temperature as recorded in the slow control viewpoint will necessarily effect the elements in the calibration viewpoint. In this case, the query facility must traverse the meta-model and determine for each effected channel the set of viewpoints it contributes to and reflect the change in those viewpoints. (Another example is detailed in section 4 and figure 8). A query facility is currently under development for CRISTAL [4].

Figure 7 shows the architecture of a meta-model based system for CMS which encompasses multiple viewpoint-specific databases (i.e. Geometry for Simulation or Event Reconstruction, Slow Controls, Alignment, Calibration, Construction). In each case data has been extracted from a general CMS meta-model (or detector description database) via the query facility. This generalised extraction facility can navigate the detector description, from a physicist-defined viewpoint, looking for specific data associated with a set of defined physics elements. The result is a totally integrated set of collaborating, but separate, databases which can be used for event storage and for determining calibration constants and can be *mined* by physicists for data from a variety of viewpoints.

In CRISTAL, physics element definitions, themselves meta-objects, are captured in the data model. They reside alongside the detector description as pre-defined 'routes' to sets of components of interest to the physicist. When specifying a new viewpoint from which data is to be extracted, the physicist can define new physics elements either from scratch or by reusing existing physics elements definitions (see figure 5). Physics elements can be nested and are queried for a specific purpose, which is captured in the data model. Data extraction from different viewpoints is supported through the capture of reusable physics elements definitions and through the provision of a query extraction facility.

In conclusion it is the definition of detector components, the tracking of assembly sequences and the gathering of physics information at stages in the execution of the assembly activities that are the main goals of the CRISTAL system. Having captured this data the ECAL construction database can be used as the information source for physics analyses, for example, in the gathering, management and presentation of data specific to calibration runs. Access to the data resident in the ECAL construction database can be generalised by the provision of a query facility which can browse the data structures in the database and present data in a format required by physicists.

4. An Example of Viewpoints: ECAL Calibration Revisited

During 1999 an ECAL Barrel module (referred to as module0 of type 2) will be built to validate the construction and assembly procedures of crystals, capsules, sub-units etc. This module will be exposed to the H4 testbeam early in 2000 for pre-calibration tests. Soon after the first production supermodule will be constructed using the knowledge accumulated from building module0 and it will also require pre-calibration. These H4 tests will need access to the ECAL construction database using the concepts of viewpoints, as defined earlier in this paper, to extract characteristics for each crystal, capsule, electronics unit etc.

For the calibration of module0, an electron beam of measured energies will be incident on a 2-dimensional array of 400 crystals (+ capsules + electronics) and beam data will be collected. Two forms of information will need to be stored: so-called parameters and data. Parameters for each readout channel include references to construction data for its crystal, capsule, electronics and Upper Level Readout (ULR), its channel number, relevant run numbers etc. and global parameters such as fibre bundle numbers also require recording. Data (in the form of events) are collected for electron beam runs (differing beam energy, table positions etc.), for slow control purposes (differing temperatures, high voltages etc.), for monitoring runs (e.g response to standard laser pulse heights), for ADC pedestal measurement and for other non-standard purposes. A run is the recording of a set of events of various types over a specific period of time and events are collections of data read from a subset of the calibration readout channels. This accumulated data is used together with the physics characteristics extracted from the construction database to calculate calibration coefficients for the readout channels (see figures 2 and 5).

Physicists will need to associate specific events recorded in the testbeam to their appropriate structures (and potentially their associated characteristics) extracted from the construction database. For example, temperature monitoring events will need to be associated with the physics elements defined for the monitoring. These physics elements thereby constitute the temperature monitoring viewpoint and are defined in terms of a set of detector components in the 'as-designed' model (as in figure 1). Similarly, high voltage events must be associated with slow control physics elements as derived from the 'as-designed' model (the slow control viewpoint) and the extraction of physics characteristics from the construction database is then carried out at a level appropriate for slow control. Figure 8 shows how the slow control viewpoint is correlated with other readout viewpoints (eg calibration and monitoring viewpoints). One member of the high voltage map (or one physics element for high voltage) controls a physical area of the ECAL Barrel detector. When energy is deposited in this area of ECAL it is recorded by a series of readout channels which themselves are the physics elements of other viewpoints (see also [8]).

For the ULR, the viewpoint needed must reflect the structure of the trigger readout where individual readout channels are grouped into strips (of 5 crystals), trigger towers (of 5 strips), ULR boards (of 4 trigger towers) and ULR crates (of 17 boards). Each viewpoint is extracted from the 'as-designed' graph for a specific purpose (e.g for beam events, for slow control events) and these viewpoints are necessarily of different structures since there are associated with physics elements (detector components) defined at different levels in the 'as-designed' detector.

5. Status and Conclusions

The CRISTAL object models are described using Unified Modelling Language, UML[6]. It is interesting to note that UML can itself be described by the Object Management Group's (OMG) Meta Object Facility [9] and more importantly is the candidate choice by OMG for describing all business models. Work is currently underway to implement the CRISTAL meta-model using OMG-standard object interfaces and using an object oriented database to support the implementation of the construction repository. A first prototype of the CRISTAL system based on the technologies outlined in this paper is due for the autumn of 1998 and the final production version by mid 1999.

The experience of using meta models and meta objects at the analysis and design phase in the CRISTAL project has been very positive. Designing the meta model separately from the runtime model has allowed the design team to provide consistent solutions to dynamic change and versioning and to support data extraction via user-defined viewpoints. The concept of using meta-data to reduce complexity and aid navigability of data resident in a database is well known [10]. Also its use in minimising the effect of schema evolution in object databases has been stated many times elsewhere [11]. In the CRISTAL project *meta-data* are used for these purposes and, in addition, *meta-models* are used to provide self-description for data and to provide the mechanisms necessary for developing a query facility to navigate multiple data models. (compare this approach to that in [12]). Using queries based on physics element definitions, data can be extracted from multiple databases and presented in user-defined viewpoints.

The CMS meta-model of figure 7 therefore acts as a repository of knowledge against which queries are issued to locate and extract data across multiple databases. Agent processes are used to 'look into' the meta-model and extract data from a user-specified viewpoint and to 'look out' from the model to correlate effects between viewpoints. The overall effect is to produce an integrated set of cooperating databases accessed through a query facilitry. The current phase of CRISTAL research aims to adopt an open architectural approach, based on a meta-model and a query facility to produce an adaptable data mining system capable of interoperating with future systems and of supporting views onto an engineering database. The meta-model approach to design reduces system complexity, provides model flexibility and can integrate multiple, potentially heterogeneous, databases into the enterprise-wide database. A first prototype for CRISTAL based on CORBA, Java and Objectivity technologies has been deployed in the autumn of 1998 [13]. The second phase of research will culminate in the delivery of a production system in 1999 supporting queries and the definition, capture and extraction of data according to physicist-defined viewpoints.

Acknowledgments

The authors take this opportunity to acknowledge the support of their home institutes – CERN and UWE. In addition, the support of P Lecoq, J-L Faure J-P Vialle and M. Pimia is greatly appreciated. The authors wish to acknowledge the input received from J-P Pansart, P Verrecchia, Ph Busson in the preparation of this note. Also the assistance of A Bazan, T Le Flour, S Lieunard, Z Kovacs, M Zsenei, S Murray, G Chevenier and F Estrella in the development of the CRISTAL prototypes is noted. During his secondment at CERN Dr R McClatchey is supported by CERN and the Royal Academy of Engineering. M Zsenei is supported by the OMFB, Hungary while working on CRISTAL

References

[1] J-M Le Goff et al., “CRISTAL - Co-operating Repositories and an Information System for Tracking Assembly Lifcycles” CERN CMS Note 1996/003.

[2] J-M Le Goff et al., “From Design to Production Control Through the Integration of Engineering Data Management and Workflow Management Systems” CERN CMS Note 1997/104

[3] J-M Le Goff et al., “Detector Construction Management and Quality Control: Establishing and Using a CRISTAL System” CERN CMS Note 1998/033

[4] F.Estrella, Z.Kovacs, J-M Le Goff, R. McClatchey & I. Willers., “The Design of an Engineering Data Warehouse Based on Meta-Object Structures”. Proc. of the Data Warehouse & Data Mining workshop at ER’98. To be published in Lecture Notes in Computer Science, Springer-Verlag.

[5] P. Maes, 1987. “Concepts and Experiments in Computational reflection”. ACM OOPSLA 1987 Proceedings. pp 147-155

[6] G. Booch, I. Jacobson & J. Rumbaugh: “UML Distilled - Applying the Standard Object Modelling Language”, Addison-Wesley Longman Inc., 1997

[8] J-P Pansart et al., “The ECAL Precalibration in H4: Users Requirements”. To appear as a CMS NOTE early in 1999.

[7] N. Baker et al., “An Object Model for product and Workflow Data Management”. Workshop proc. At the 9^{th} Int. Conference on Database & Expert System Applications. Vienna, Austria August 1998.

[9] Object Management Group Publications, Common Facilities RFP-5 Meta-Object Facility TC Document cf/96-02-01 R2, 5/13/96.

[10] Selected papers from the proceedings of the 1^{st} IEEE Metadata conference, Silver Spring, Maryland, 1996.

[11] W. Kim., “Modern Database Systems”. ACM Press / Addison Wesley Publishers, 1995

[12] B. Kerherve & A. Gerbe., “Models for Metadata or Metamodels for Data?”. Proc of the 2^{nd} IEEE Meta-Data Conference, Silver Spring, Maryland, 1997.

[13] A. Bazan et al., “The Use of Production Management Techniques in the Construction of Large Scale Physics Detectors”. Accepted at the IEEE Nuclear Science Symposium and Medical Imaging conference, Toronto, Canada. November 1998.

Figures

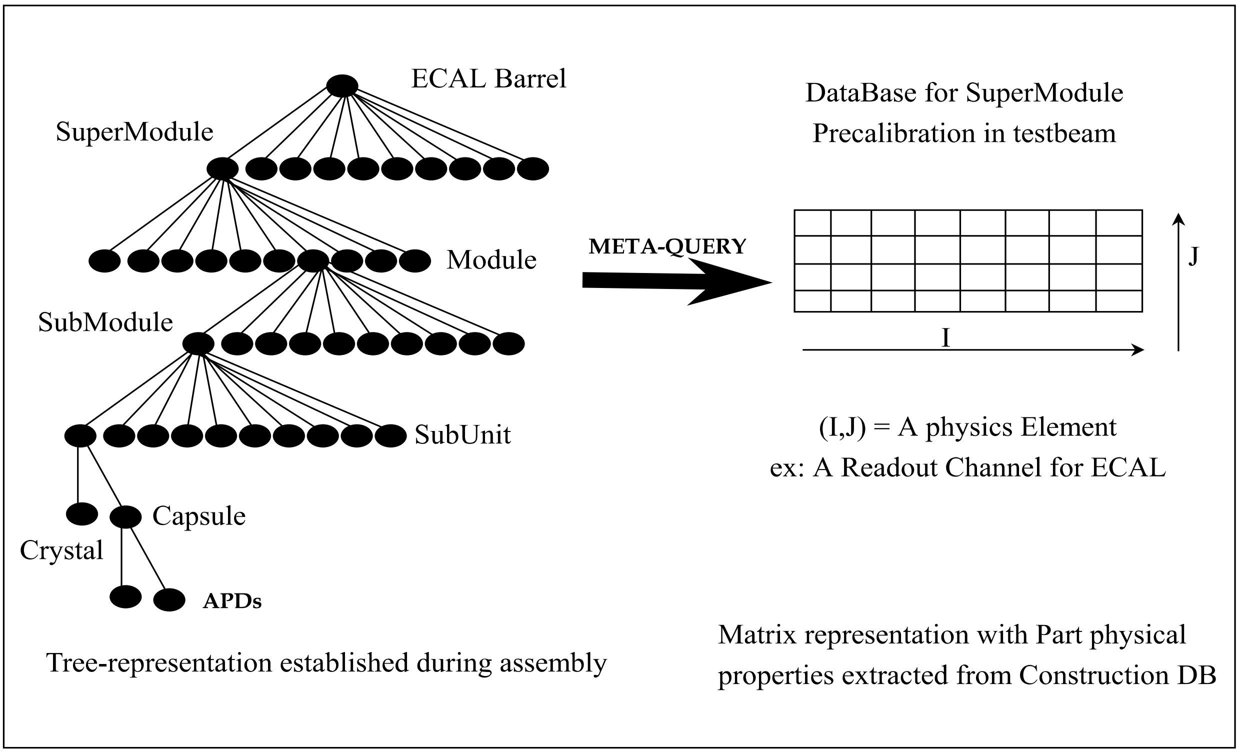

Figure 1: Deriving the calibration viewpoint (matrix of 'physics elements') from the construction database.

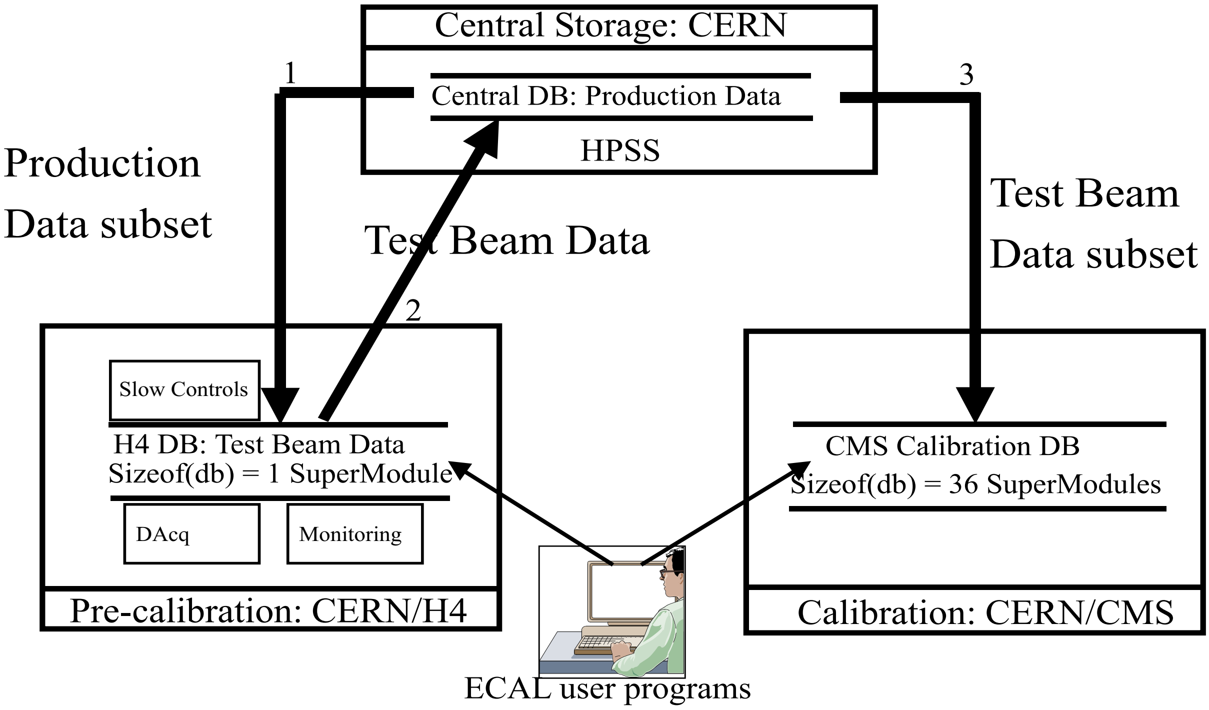

Figure 2: The extraction of physics data from the ECAL construction database.

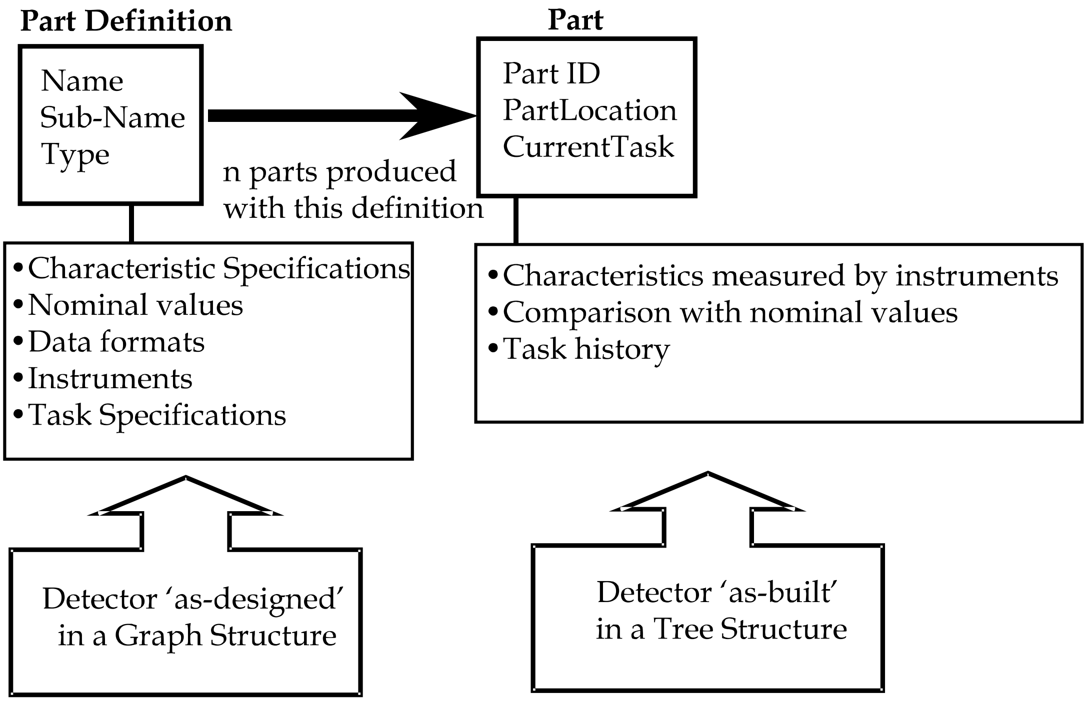

Figure 3: Meta-objects: the basis of the CRISTAL self-describing meta-model.

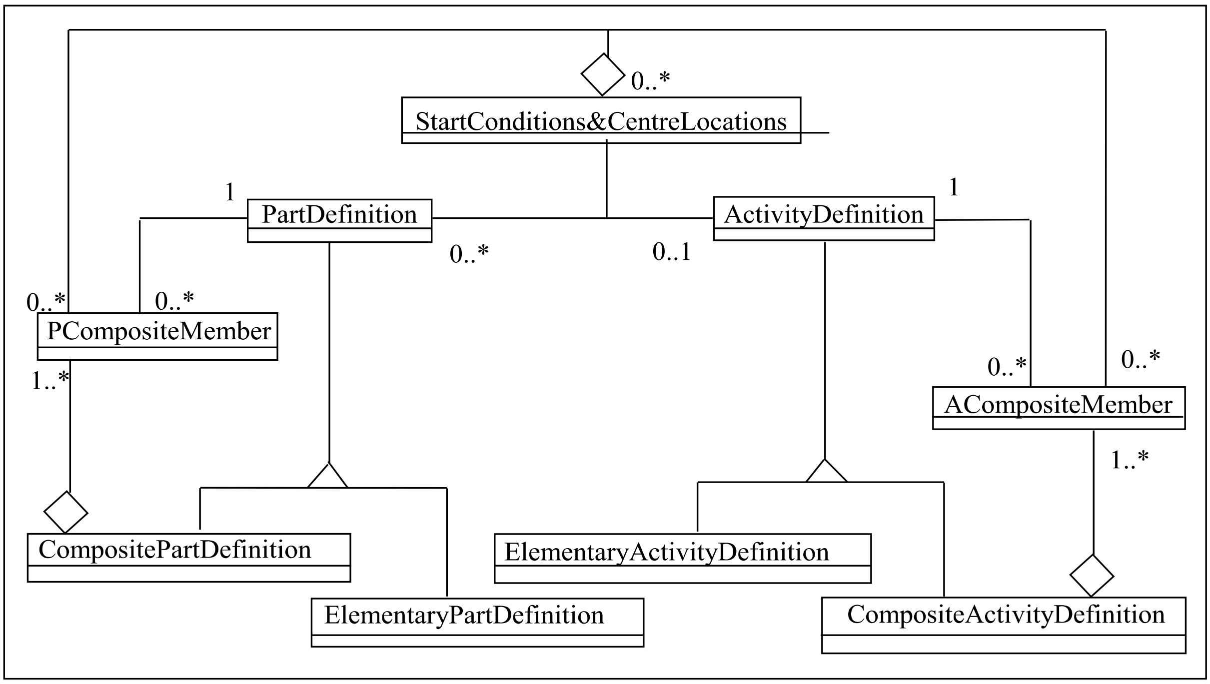

Figure 4: A simplified subset of the CRISTAL UML object model.

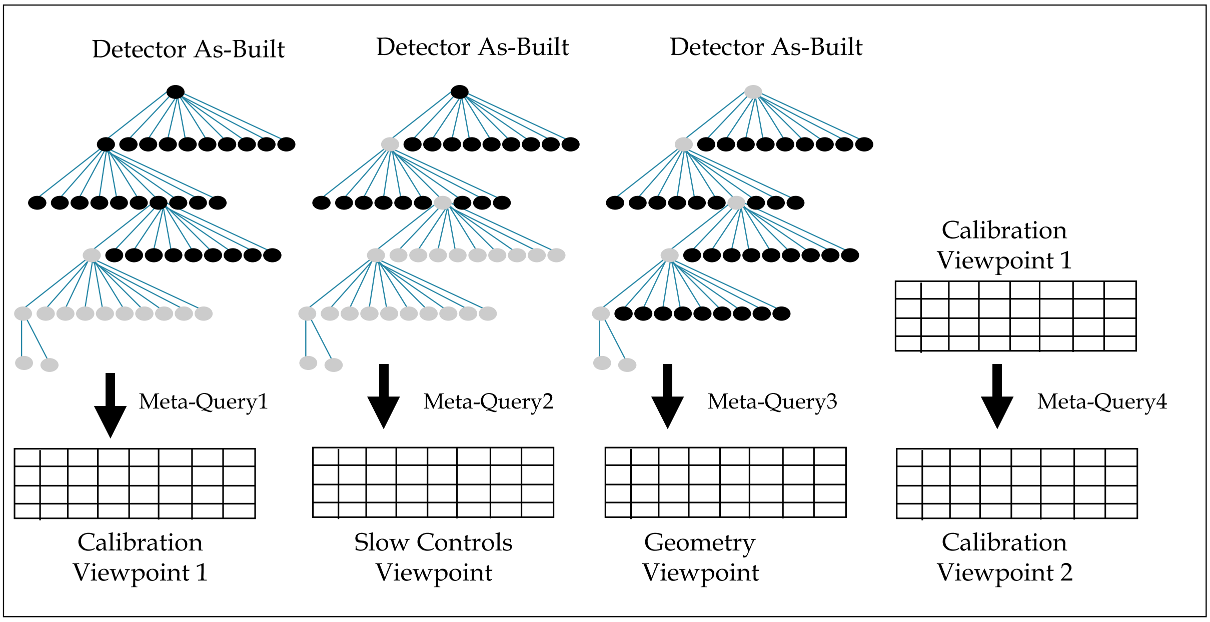

Figure 5: Multiple Viewpoints and Reuse of Viewpoints

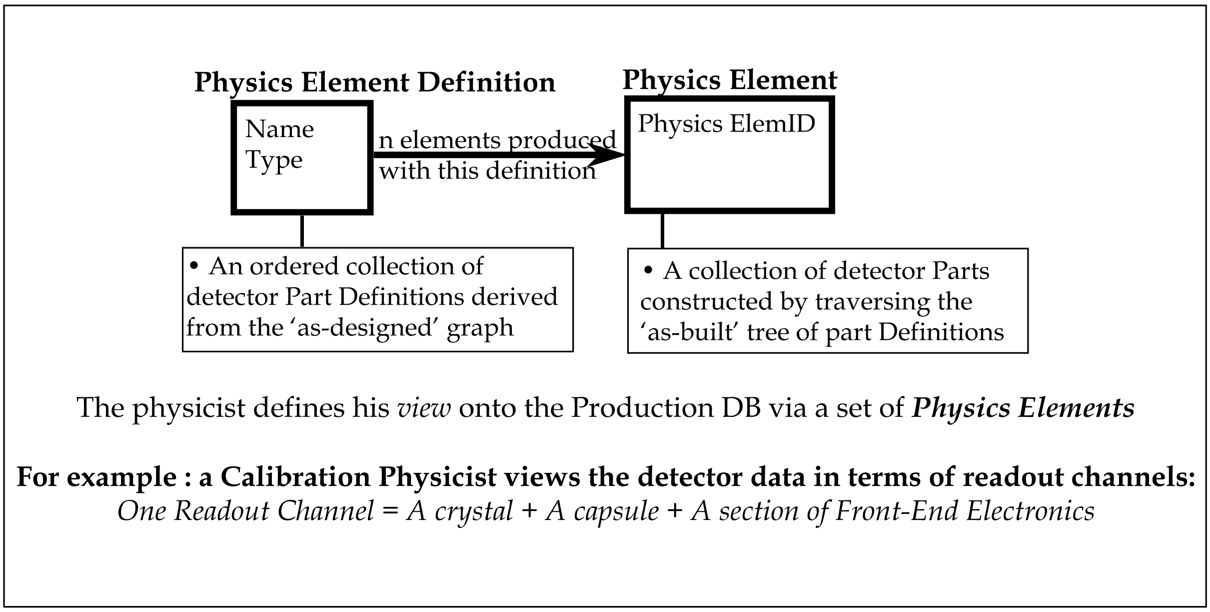

Figure 6: The definition of physics elements for a Calibration viewpoint.

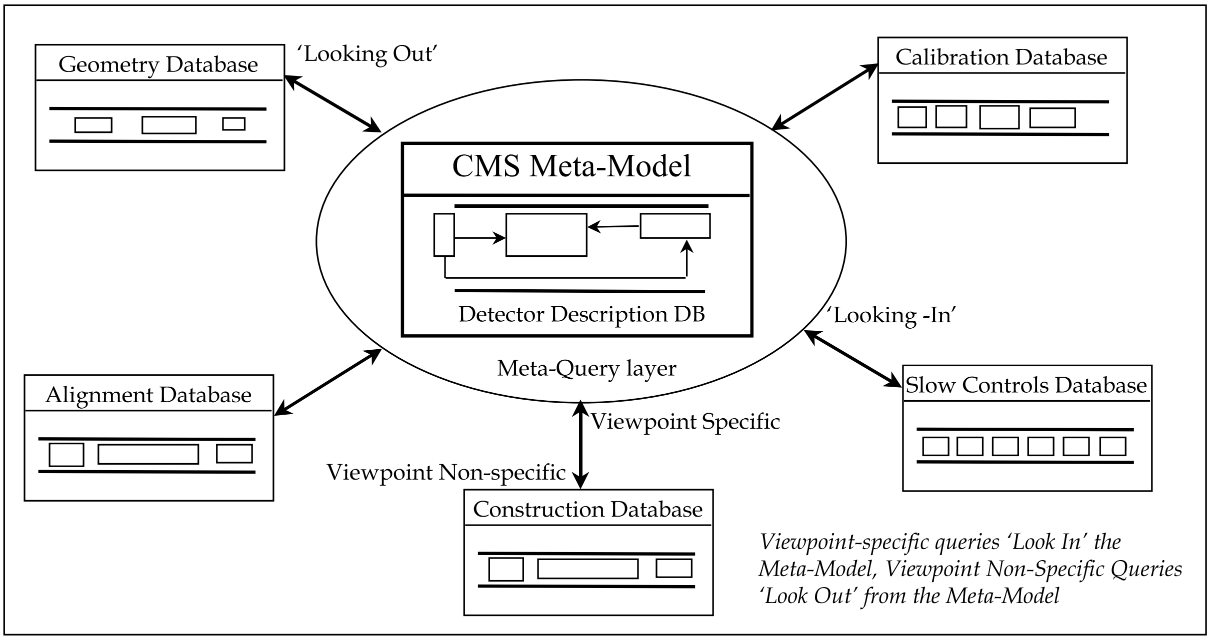

Figure 7: A generalised detector description database and query facility for CMS.

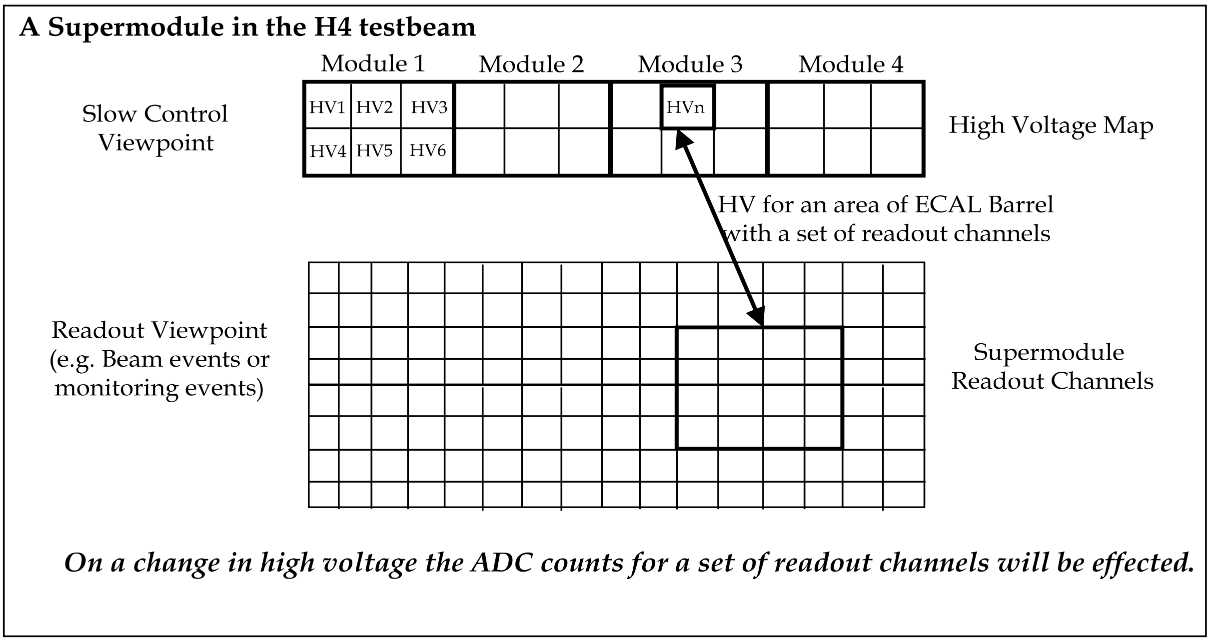

Figure 8: An example of the correlation between viewpoints, slow control events (e.g a change in high voltage) can effect the physics elements (e.g. readout channels) of other viewpoints